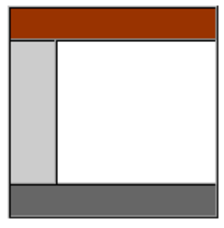
## Лабораторная работа №1

## Основы блочной верстки страниц

## Задание 1. Две колонки в резиновом дизайне

Дизайн CSS две колонки, пожалуй, самый популярный. Создать страницу с таким расположением колонок можно двумя способами, рассмотрим каждый отдельно.

Рассмотрим пример, в котором присутствуют следующие блоки, изображенные на *рис. 1*: шапка (бордовый), меню (светло-серый), контент (белый), подвал (темно-серый).

[](http://labs.org.ru/wp-content/uploads/2016/12/1-65.png)

*Рис. 1. Резиновый дизайн сайта в две колонки*

Левую колонку выполним с фиксированным размером, а правая колонка будет занимать остальную ширину окна, что и обеспечит «резиновость» дизайна.

Основные свойства

1. «Разбиваем» все основные элементы страницы на блоки следующим образом:

<div id="top">Заголовок сайта</div>

<div id="menu">

Ссылка 1<br>

Ссылка 2<br>

Ссылка 3<br>

Ссылка 4<br>

</div>

<div id="content">

<h1>Заголовок</h1>

<p>Контент. Дизайн CSS две колонки, пожалуй, самый популярный. Создать страницу с таким расположением колонок можно двумя способами, рассмотрим каждый отдельно.

</p>

<p>С помощью CSS выполняется логическое форматирование html-документа, позволяющее разделить содержание и оформление html-документа Использование CSS рекомендовано к применению WWW-консорциумом.

</p>

</div>

<div id="bottom">

Подвал

</div>

2. Для левой колонки задаем свойства, обязательные при использовании резинового дизайна: float и width ¶1 вариант:

|  |
| --- |
| #menu{  float: left;  width: 20%;  } |

2 вариант:

|  |
| --- |
| #menu{  float: left;  width: 200px;  } |

3. Для правой колонки обязательным является свойство, определяющее внешний отступ от левого края окна браузера (margin-left). Этот отступ должен соответствовать ширине левой колонки + 1px или 1%:

¶1 вариант:

|  |
| --- |
| #content{  margin-left: 21%;  } |

2 вариант:

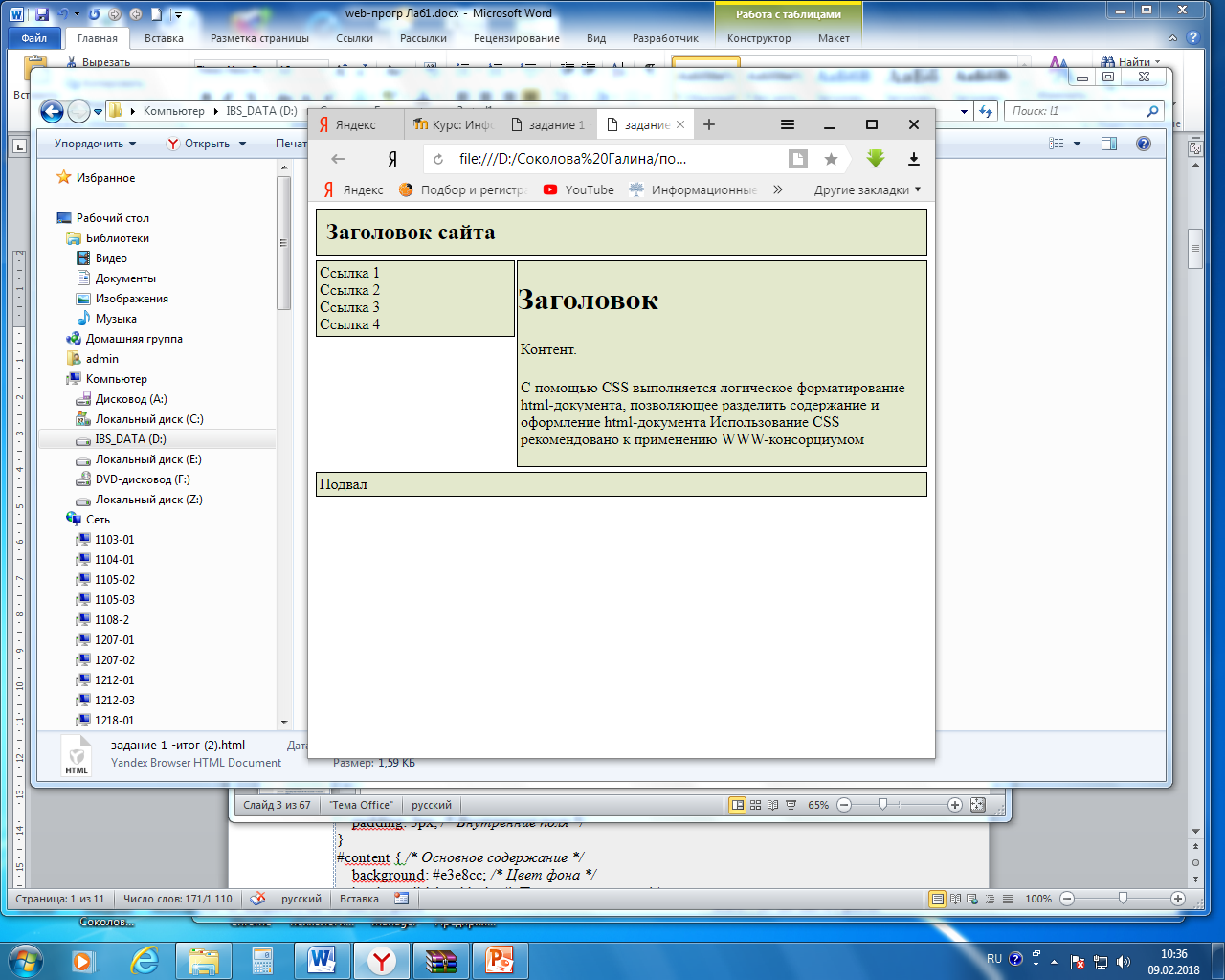
|  |
| --- |
| #content{  margin-left: 210px;  } |

Все остальные свойства зависят от оформления сайта.

Весь код CSS будет выглядеть так:

|  |
| --- |
| #top { */\* Шапка с заголовком страницы \*/*  background: #e3e8cc; */\* Цвет фона \*/*  border: solid 1px black; */\* Параметры рамки \*/*  padding: 10px 0 10px 10px; */\* Поля вокруг текста \*/*  margin-bottom: 5px; */\* Расстояние между шапкой и колонками \*/*  font-size: 24px; */\* Размер шрифта \*/*  font-weight: bold; */\* Жирное начертание \*/*  }  #menu { */\* Меню сайта \*/*  width: 20%; */\* Ширина меню \*/*  background: #e3e8cc; */\* Цвет фона \*/*  border: solid 1px black; */\* Параметры рамки \*/*  float: left; */\* Состыковка со слоем контента по горизонтали \*/*  padding: 3px; */\* Внутренние поля \*/*  }  #content { */\* Основное содержание \*/*  background: #e3e8cc; */\* Цвет фона \*/*  border: solid 1px black; */\* Параметры рамки \*/*  margin-left: 21%; */\* Отступ слева с учетом ширины левой колонки \*/*  margin-bottom: 5px */\* Отступ снизу \*/*  }  #content p{  padding: 3px; */\* Внутренние поля контента\*/*  }  #bottom { */\* Подвал \*/*  background: #e3e8cc; */\* Цвет фона \*/*  border: solid 1px black; */\* Параметры рамки \*/*  padding: 3px; */\* Поля вокруг текста \*/*  } |

Результат:



*Рис. 2. Левая колонка с фиксированной шириной в резиновом дизайне*

Рассмотрим пример, когда фиксированной шириной обладает не левая, а правая колонка.

## Задание 2. Правая колонка с фиксированной шириной: основные свойства

Код html-структуры остается прежним, меняются лишь CSS-свойства.

1. Для левой колонки задаем свойства, обязательные при использовании резинового дизайна: float и width

¶1 вариант:

|  |
| --- |
| #menu{  float: right;  width: 20%;  } |

2 вариант:

|  |
| --- |
| #menu{  float: right;  width: 200px;  } |

1. Для правой колонки задаем свойство, определяющее внешний отступ от правого края окна браузера (margin-right). Этот отступ должен соответствовать ширине левой колонки + 1px или 1%:

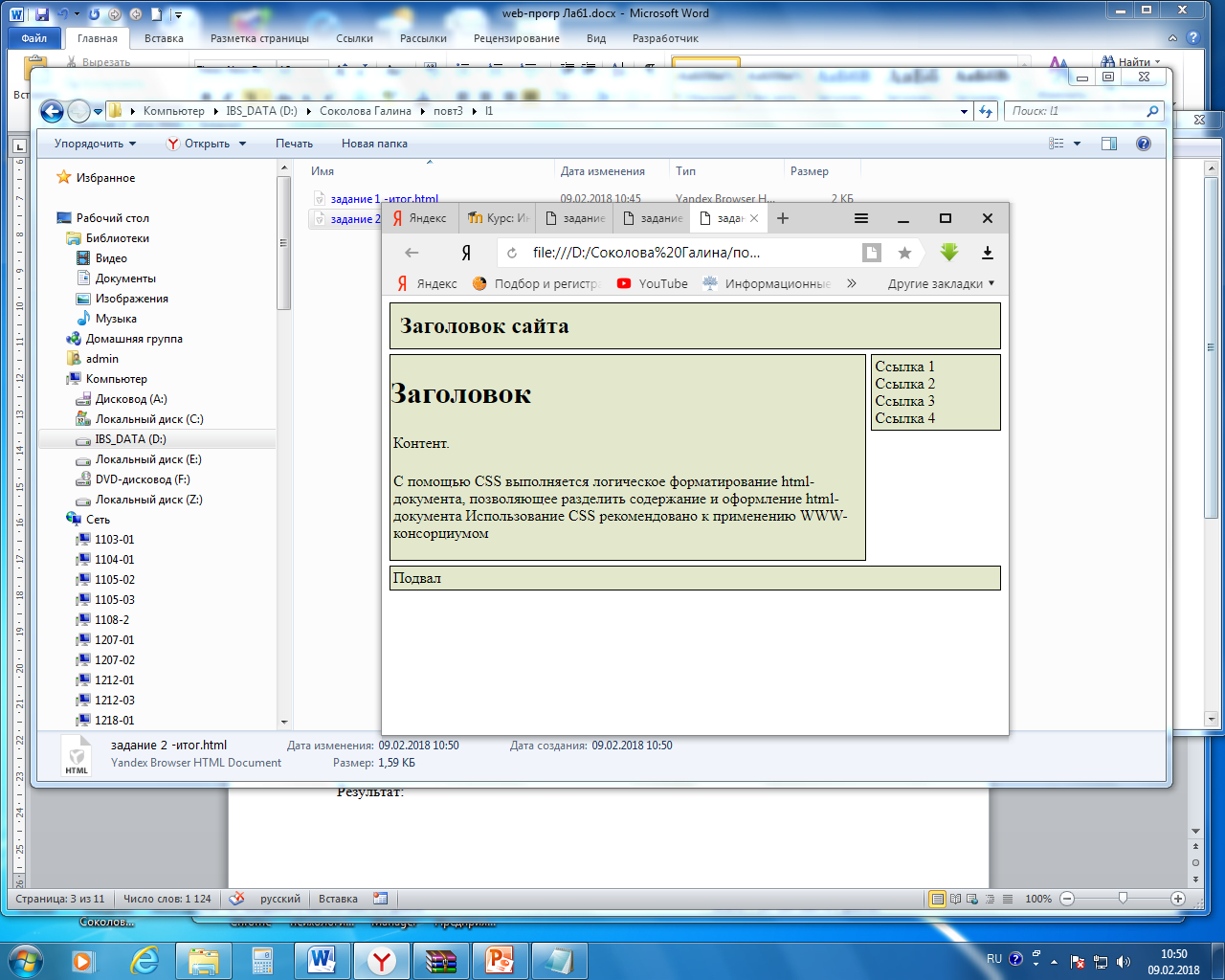
¶1 вариант:

|  |
| --- |
| #content{  margin-right: 21%;  } |

2 вариант:

|  |
| --- |
| #content{  margin-right: 210px;  } |

Результат:



*Рис. 3. Правая колонка с фиксированной шириной в резиновом дизайне*

## Задание 3. Применение позиционирования в резиновом дизайне

При использовании данного варианта дизайна для левой или правой колонки устанавливается абсолютное позиционирование с заданием координат.

Например:

Левая колонка:

|  |
| --- |
| position:absolute;  width: ...px;  left: ...px;  top: ...px; |

Правая колонка:

|  |
| --- |
| margin-left: ...px; */\* отступ слева, равный ширине первой колонки \*/* |

«Резиновость» достигается в данном случае за счет правой колонки, у которой нет определенной ширины.

Пример: Создать веб-страницу в формате резинового дизайна. Для левой колонки задать абсолютное позиционирование с координатами от левого и верхнего края окна браузера

¶Выполнение:

1. Расположим основные html-элементы страницы следующим образом:

|  |
| --- |
| <div id="col1">  <p>  Ссылка 1<br>  Ссылка 2<br>  Ссылка 3<br>  Ссылка 4<br>  </p>  </div>  <div id="col2">  <p>Контент. Дизайн CSS две колонки, пожалуй, самый популярный. Создать страницу с таким расположением колонок можно двумя способами, рассмотрим каждый отдельно.  </p>  <p>С помощью CSS выполняется логическое форматирование html-документа, позволяющее разделить содержание и оформление html-документа Использование CSS рекомендовано к применению WWW-консорциумом.  </p>  </div> |

2. Для левой колонки задаем свойства, обязательные при использовании позиционирования в резиновом дизайне — position, width, left и top:

|  |
| --- |
| #col1 { */\* Левая колонка \*/*  position: absolute; */\* Абсолютное позиционирование \*/*  width: 190px; */\* Ширина блока\*/*  left: 0px; */\* Координата от левого края окна \*/*  top: 55px; */\* Координата от верхнего края окна \*/*  background: #800000; */\* Цвет фона левой колонки \*/*  padding: 5px; */\* Поля вокруг текста \*/*  } |

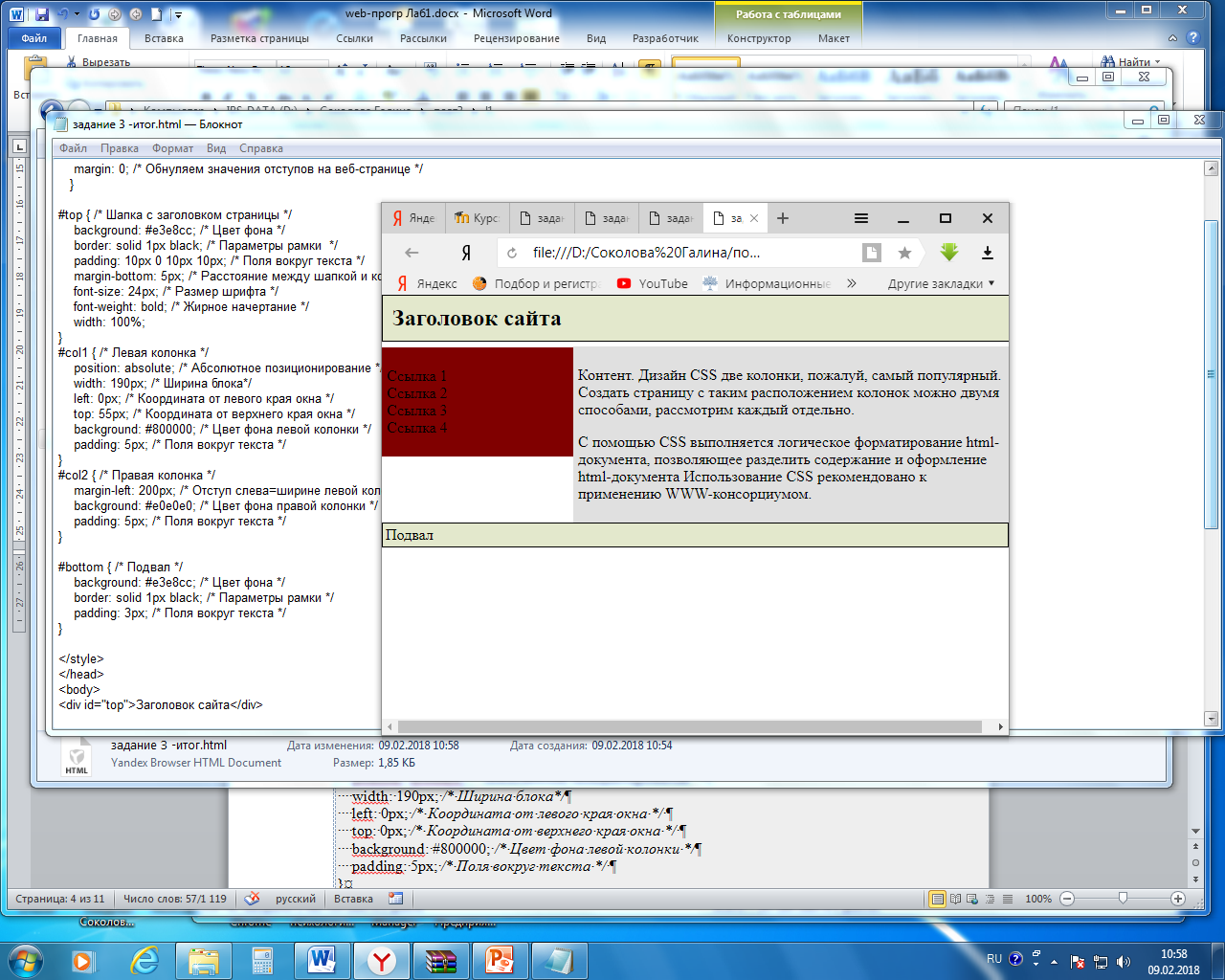
3. Задаем свойства для правой колонки, обязательные при использовании позиционирования в резиновом дизайне — margin-left:

|  |
| --- |
| #col2 { */\* Правая колонка \*/*  margin-left: 200px; */\* Отступ слева=ширине левой колонки + padding \*/*  background: #e0e0e0; */\* Цвет фона правой колонки \*/*  padding: 5px; */\* Поля вокруг текста \*/*  } |

4. Удаляем лишние отступы браузера по умолчанию:

|  |
| --- |
| body {  padding: 0; */\* Отступы для браузера Opera \*/*  margin: 0; */\* Обнуляем значения отступов на веб-странице \*/*  } |

Результат:



*Рис. 4. Резиновый дизайн сайта с позиционированием*

## Задание 4. Одинаковая высота колонок

На *рис. 4* результат показывает, что колонки имеют неодинаковую высоту. Для того, чтобы колонки приобрели одинаковую высоту есть несколько вариантов использования дополнительных свойств, одним из которых является добавление границы (border).

1. Обе колонки заключаются в слой-контейнер:

|  |
| --- |
| <div id="container">  <div id="col1">  <p>  Ссылка 1<br>  Ссылка 2<br>  Ссылка 3<br>  </p>  </div>  <div id="col2">  <p>Контент. Дизайн CSS две колонки, пожалуй, самый популярный. Создать страницу с таким расположением колонок можно двумя способами, рассмотрим каждый отдельно.  </p>  <p>С помощью CSS выполняется логическое форматирование html-документа, позволяющее разделить содержание и оформление html-документа Использование CSS рекомендовано к применению WWW-консорциумом.  </p>  </div>  </div> |

2. Для левой колонки устанавливаем ширину, абсолютное позиционирование, координаты от верхнего левого угла браузера, внешний отступ слева отрицательного значения (!), равный ширине блока, и внутреннее поле слева, равное также ширине блока:

|  |
| --- |
| #col1 { */\* Левая колонка \*/*  position: absolute; */\* Абсолютное позиционирование \*/*  width: 190px; */\* Ширина слоя \*/*  left: 0px; */\* Положение от левого края окна \*/*  top: 0px; */\* Положение от верхнего края окна \*/*  margin-left: -200px; */\* внешний левый отступ отрицательного значения \*/*  padding-left: 200px; */\* внутренне левое поле \*/*  background: #800000; */\* Цвет фона левой колонки \*/*  } |

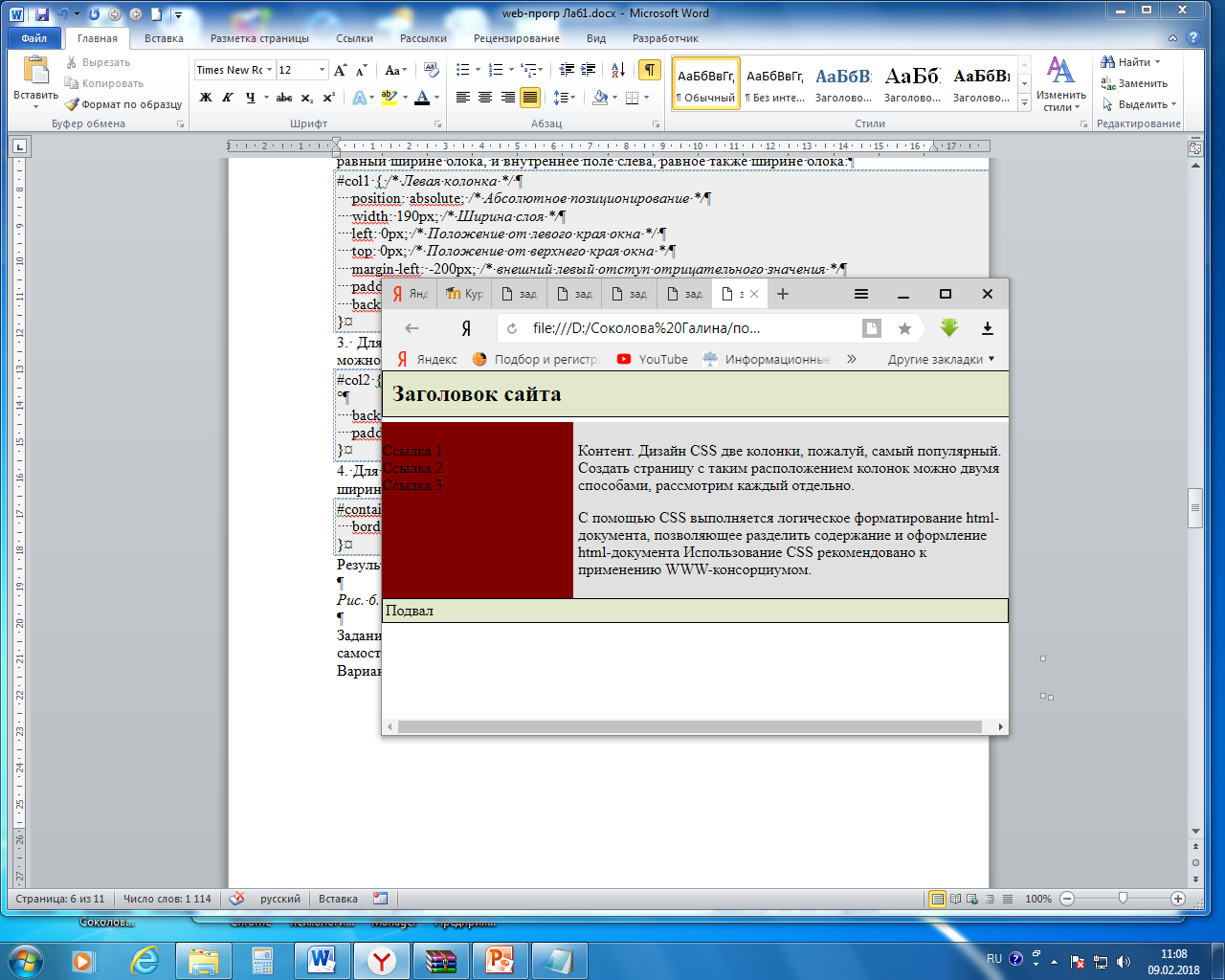
3. Для правой колонки никакие свойства позиционирования не устанавливаются, но можно задать свойства для оформления колонки:

|  |
| --- |
| #col2 { */\* Правая колонка \*/*    background: #e0e0e0; */\* Цвет фона правой колонки \*/*  padding: 5px; */\* Поля вокруг текста \*/*  } |

4. Для слоя контейнера необходимо задать границу, ширина которой должна совпадать с шириной левой колонки, а цвет — с цветом левой колонки:

|  |
| --- |
| #container {  border-left: 200px solid #800000;  } |

Результат:



*Рис. 6. Колонки одинаковой высоты*

## ****Задание 5. Создание трех страничного сайта****

На основе любого из шаблонов выполненных в заданиях 1-4 создать трех страничный сайт следующей структуры:

**Главная страница**

**index.html**

**Текст 1**

**text1.html**

**Текст 2**

**text2.html**

Все html-страницы должны находиться в корневом каталоге вашего сайта, все рисунки – в подкаталоге images.

Текст для страницы text1.html взять из файла text1.doc.

Текст для страницы text2.html взять из файла text2.doc.

Отредактировать страницу text1.html.

Выполнить разметку.

Внутри страницы организовать переход на метки. Основные разделы текста, а именно:

Введение

Основные поисковые машины

Особенности поисковых машин

должны быть в начале страницы в виде гиперссылок на определенную позицию в тексте. В случае, когда используются переходы внутри текущей страницы, на ней должны быть расставлены метки:

<A name=”Метка”> </A>

Для перехода к метке используется ссылка по следующему шаблону:

Текст подсказки <A href=”#Метка”> Текст для щелчка </A>

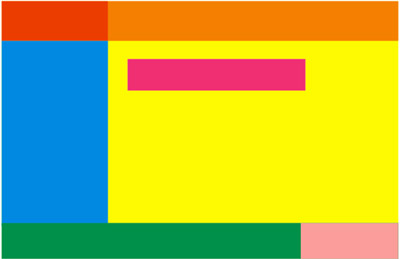
Добавить картинки на страницы сайта.

Выполнить валидацию HTML и CSS кода.

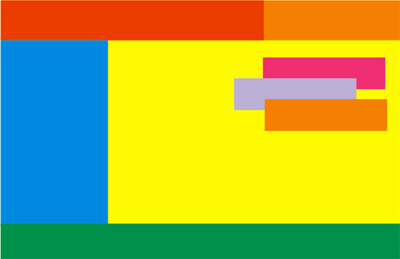
## ****Задание для самостоятельного выполнения****

Методом блочной верстки создать композицию web-страницы. Размеры выбрать самостоятельно с соблюдением пропорций на рисунке.

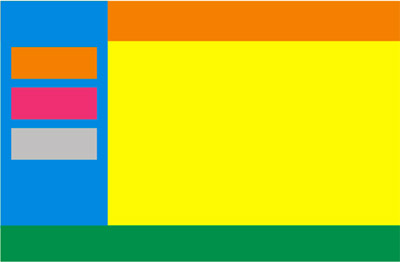
**Вариант 1**



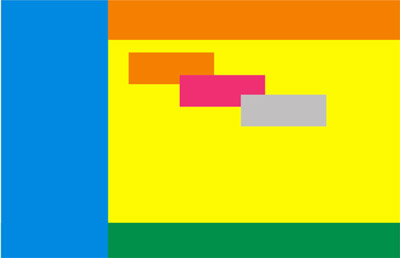
**Вариант 2**



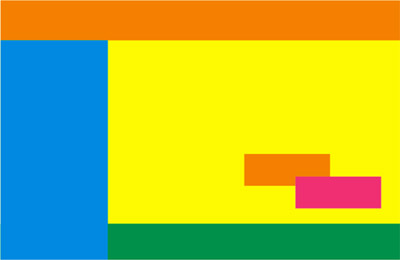
**Вариант 3**



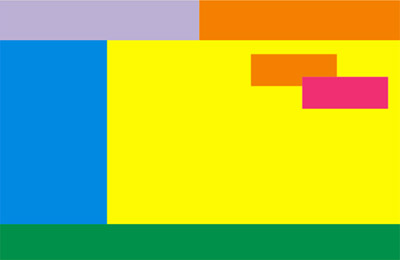
**Вариант 4**



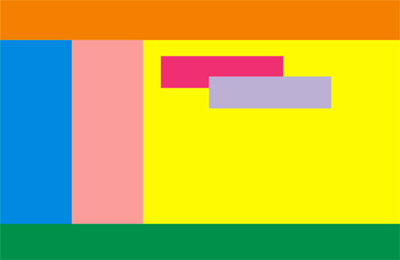
**Вариант 5**



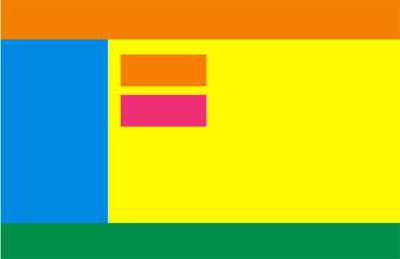
**Вариант 6**



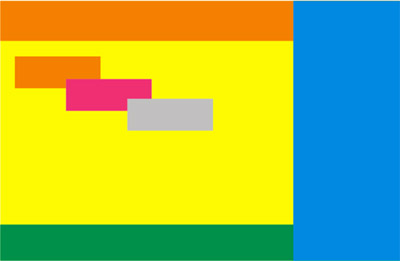
**Вариант 7**



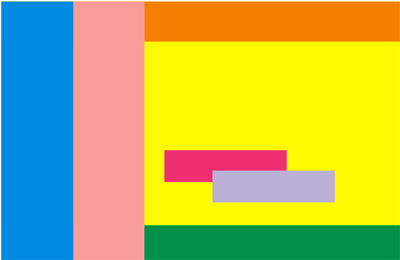
**Вариант 8**



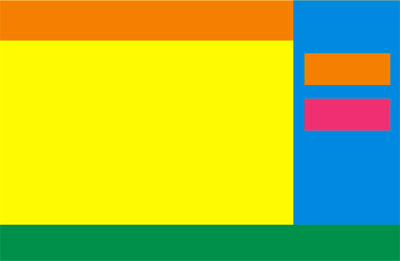
**Вариант 9**



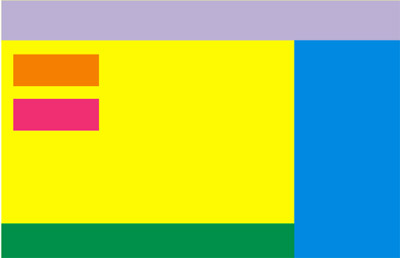
**Вариант 10**



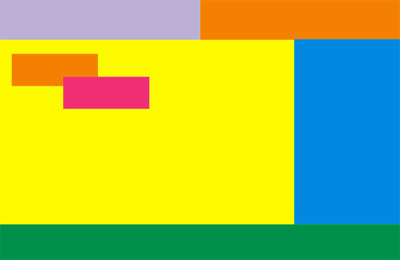
**Вариант 11**



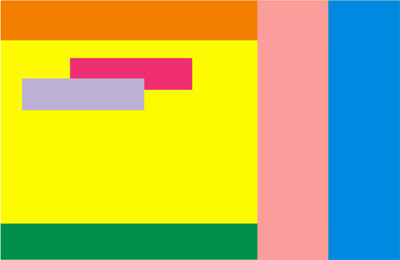
**Вариант 12**



**Вариант 13**



**Вариант 14**



**Вариант 15**

